

En los apuntes del itinerario 2 existe un ejemplo desarrollado completamente.

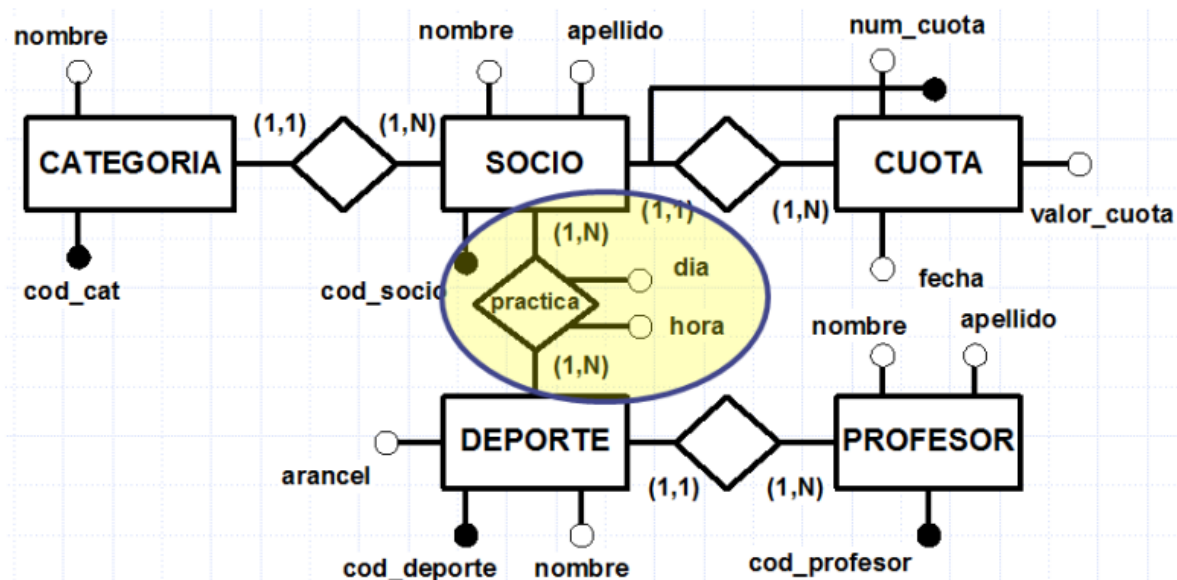
La consigna del mismo versa:

*“El sistema administra la información relacionada con un club deportivo. Los socios pueden ser de distintas categorías; ésta depende de su antigüedad. La cuota social, tiene dos componentes, consta de un básico que depende de la categoría del socio más un plus que depende del deporte elegido. Se desea tener, además, información sobre los deportes que practica cada socio, con la siguiente información: deporte, día y hora de práctica, profesor y arancel. Un socio puede practicar distintos deportes. Cada profesor trabaja en un sólo deporte pero, por supuesto, en cada deporte trabaja más de un profesor”*

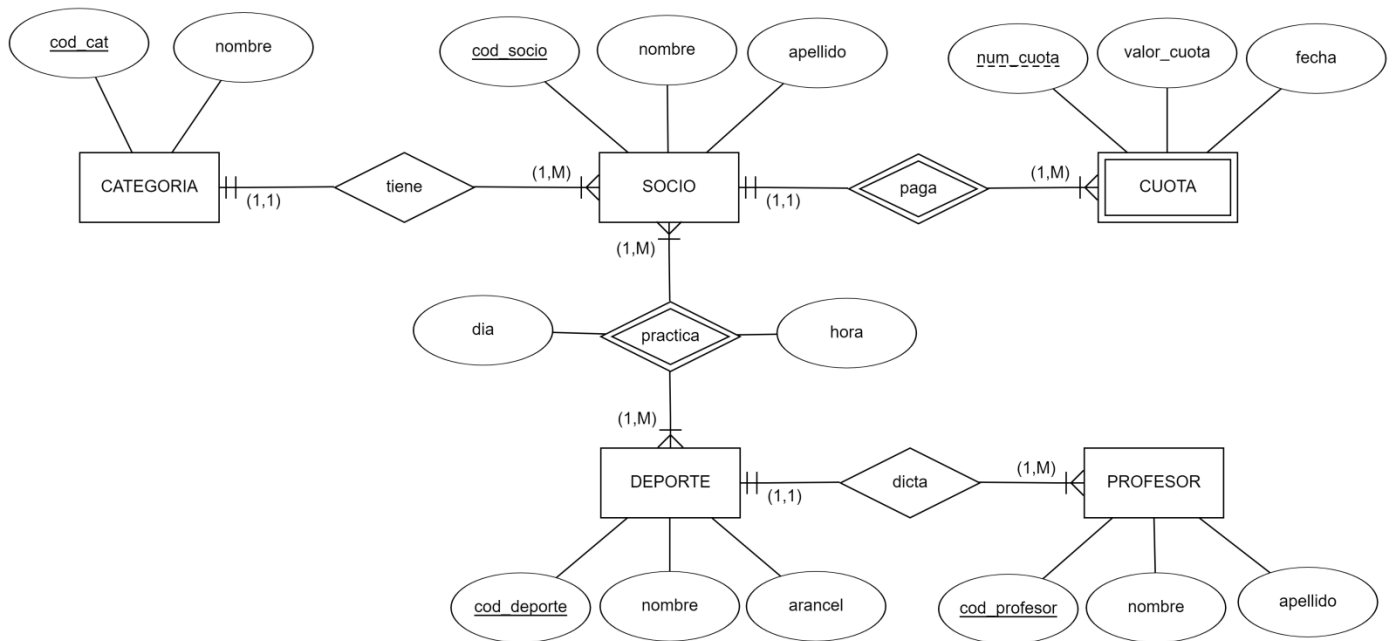
## PASO A PASO (CON EXPLICACIONES) CÓMO SE REALIZÓ EL TRABAJO:

Primero había que “trasladar” el ejemplo original (en simbología de Batini) a “la clásica” (que es la que vimos en clase y que figura en la gran mayoría de los libros):

### EJEMPLO - PRÁCTICO

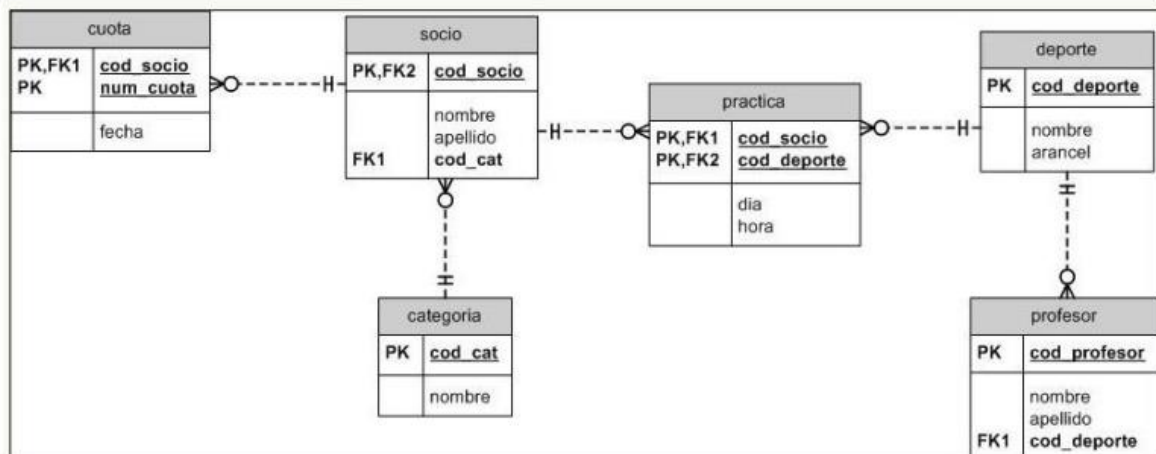


La conversión implicó comprender conceptos que uno cree que los entiende hasta que tiene que hacer una práctica:

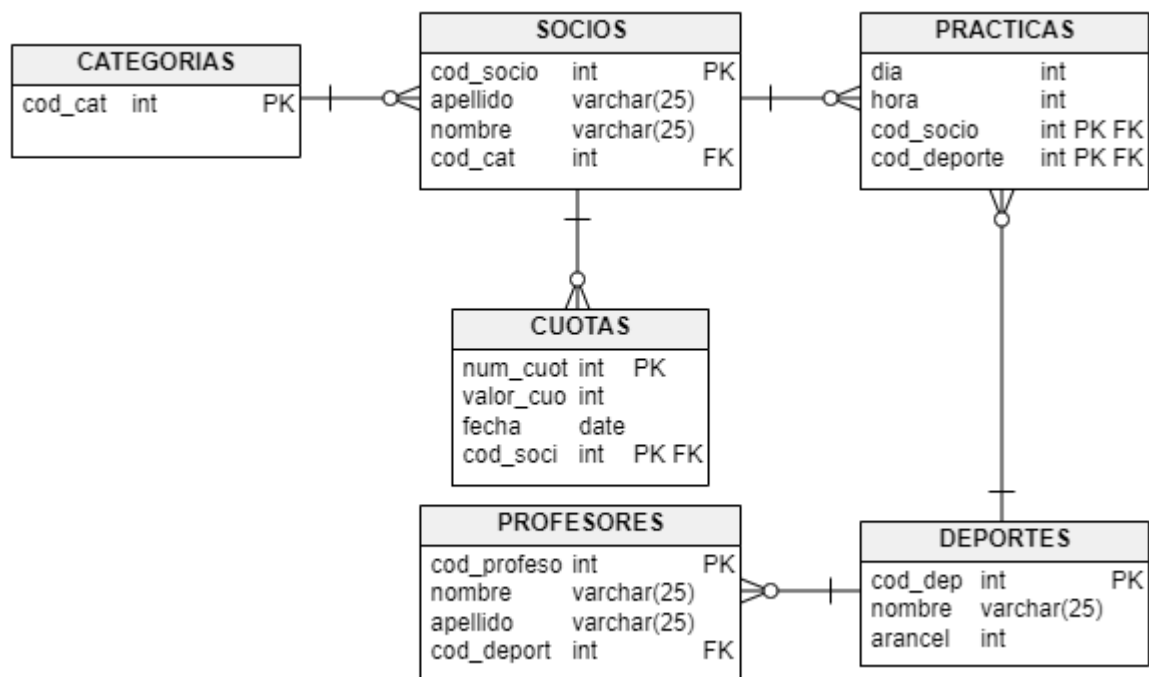


En el ejemplo original se obtiene el siguiente diagrama lógico (el cual debe salir del diagrama conceptual, obviamente):

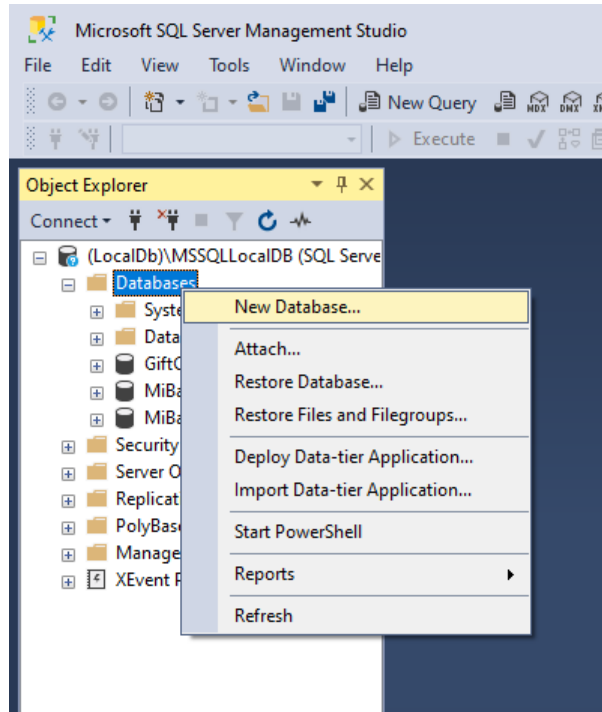
## EJEMPLO - PRÁCTICO



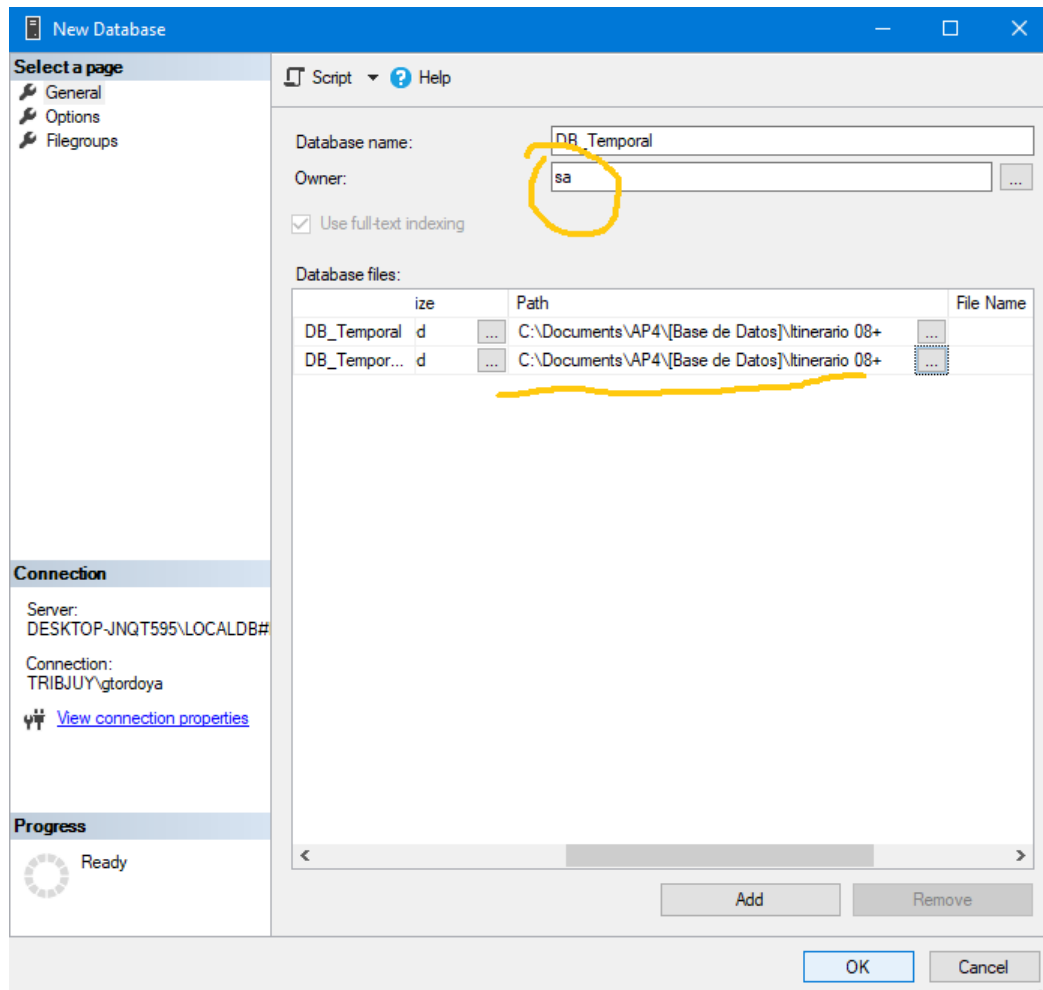
Trabajando el modelo propio, conseguí ese mismo esquema (hay una diferencia en la tabla cuota, donde desaparece el valor cuota, pero luego, en las láminas siguientes, reaparece el concepto, así que lo di por bueno):



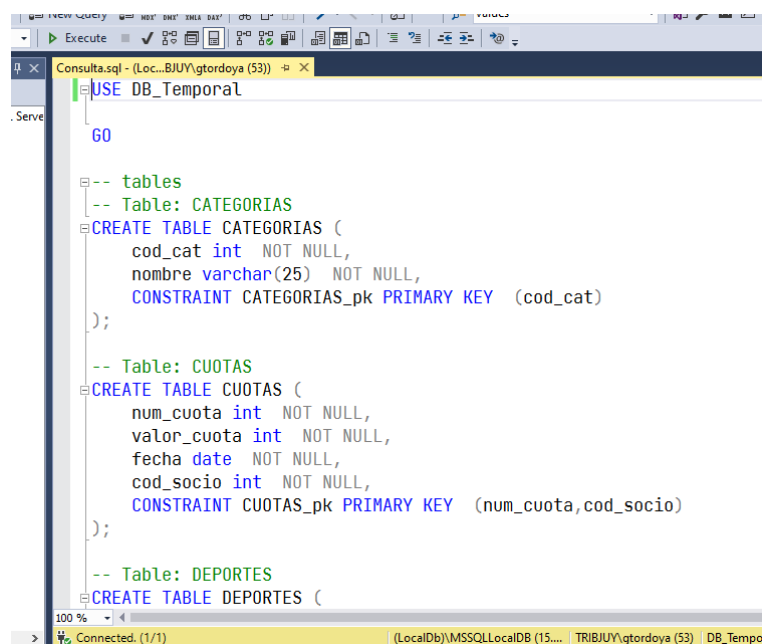
Una vez conseguido reproducir el diseño, era hora de trabajar con el SSMS para concretar la base de datos. Así que empecé creando una a la que llamé DB\_Temporal (porque no tiene otra finalidad que la de estas capturas):

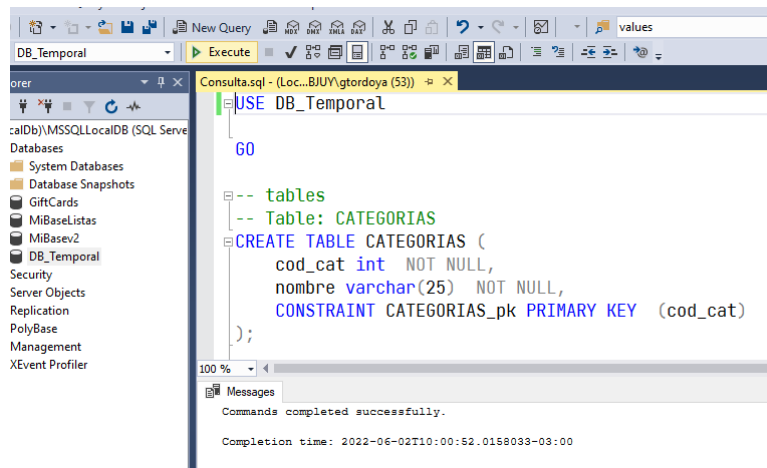


De lo que se ha aprendido hasta ahora, se hicieron esos “pellizcos” (ubicación de los archivos, permisos) que luego, cuando salta el error, se hacen un poco más difícil arreglarlos:

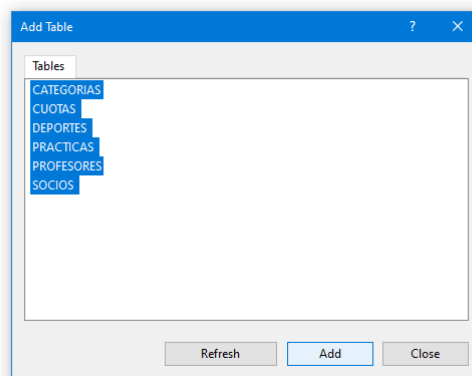
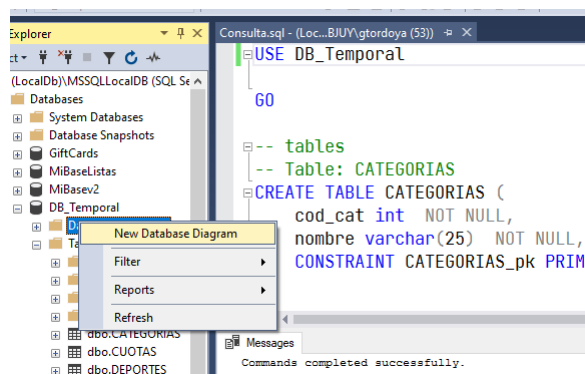
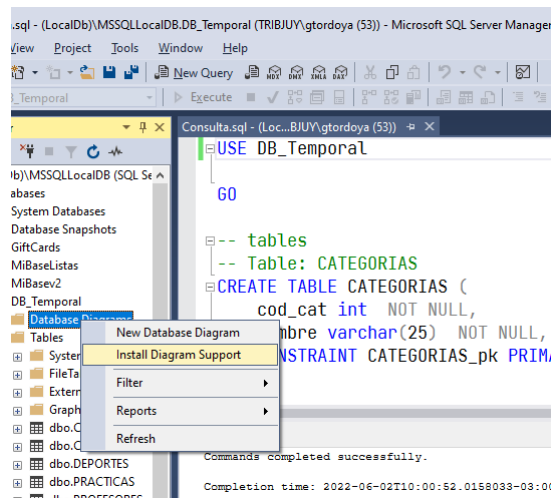


Ya teniendo creada la BD, era hora de correr el SQL que se generó a partir del diagrama lógico (varios programas lo hacen automáticamente, la consulta SQL que crea tablas y relaciones no la hice yo):

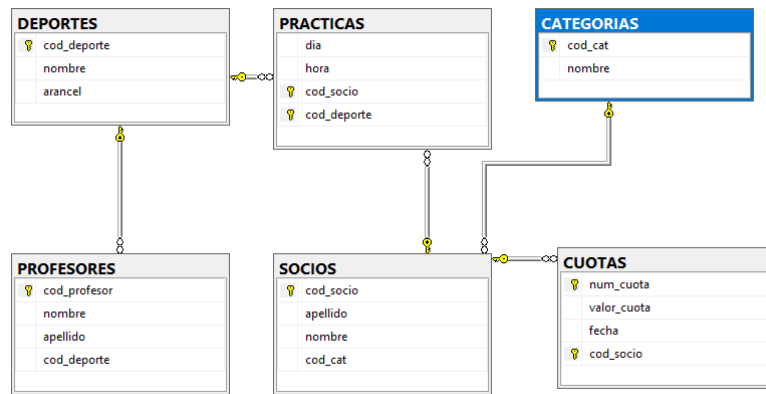




Momento de comparar si el modelo físico se corresponde al modelo lógico, así que le pidamos a SSMS que genere el diagrama. Para eso, instalemos las herramientas necesarias:



Y... ¡ualá!, coincide con el diagrama lógico:



El ejercicio original tiene algunas consultas. Para reproducir esas consultas, fue necesario cargar algunos datos, más o menos como para obtener resultados similares a los del ejemplo. Corremos las 4 consultas del ejemplo original para comparar si los resultados se asemejan.

Consulta 1:

```
USE DB_Temporal
GO

--Listado de socios por categoría

SELECT socios.nombre, socios.apellido, categorias.nombre
FROM socios, categorias
WHERE socios.cod_cat = categorias.cod_cat
ORDER BY categorias.nombre;
```

	nombre	apellido	nombre
1	Mario	Perez	Cadete
2	Jorge	Rodriguez	Cadete
3	Juana	Rey	Juvenil

Consulta 2:

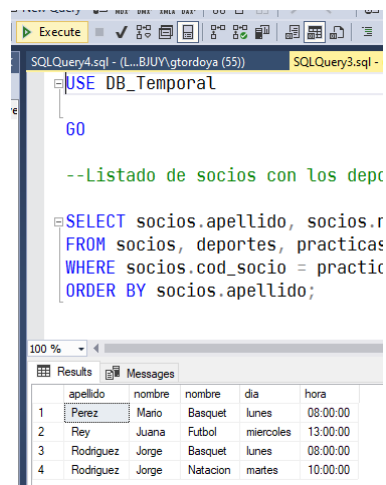
```
USE DB_Temporal
GO

--Listado de socios y sus cuotas pagas

SELECT socios.cod_socio, socios.apellido, cuotas.num_cuota
FROM socios, cuotas
WHERE socios.cod_socio = cuotas.cod_socio
ORDER BY socios.cod_socio, cuotas.num_cuota;
```

	cod_socio	apellido	num_cuota	fecha
1	1	Perez	1	2005-01-01
2	1	Perez	2	2005-01-02
3	2	Rodriguez	1	2005-01-01
4	2	Rodriguez	2	2005-01-02

### Consulta 3:



The screenshot shows a SQL query window with the following code:

```
USE DB_Temporal
GO

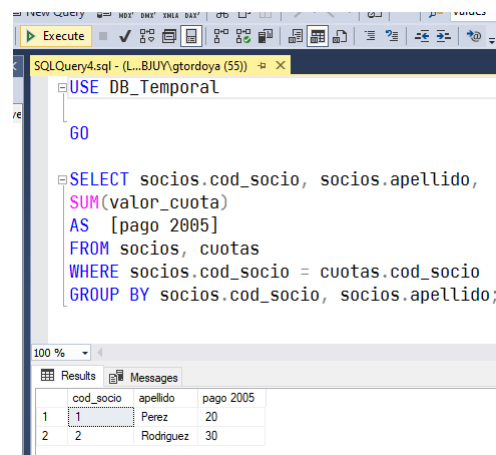
--Listado de socios con los deports

SELECT socios.apellido, socios.r
FROM socios, deportes, practicas
WHERE socios.cod_socio = practica
ORDER BY socios.apellido;
```

The results pane shows the following data:

	apellido	nombre	nombre	dia	hora
1	Perez	Mario	Basquet	lunes	08:00:00
2	Rey	Juana	Futbol	miercoles	13:00:00
3	Rodriguez	Jorge	Basquet	lunes	08:00:00
4	Rodriguez	Jorge	Natacion	martes	10:00:00

### Consulta 4:



The screenshot shows a SQL query window with the following code:

```
USE DB_Temporal
GO

SELECT socios.cod_socio, socios.apellido,
SUM(valor_cuota)
AS [pago 2005]
FROM socios, cuotas
WHERE socios.cod_socio = cuotas.cod_socio
GROUP BY socios.cod_socio, socios.apellido;
```

The results pane shows the following data:

	cod_socio	apellido	pago 2005
1	1	Perez	20
2	2	Rodriguez	30

Y sí, están bien, coinciden con el ejemplo original.

De todo lo propuesto, ese ejemplo me pareció el más completo y el que más cosas enseñaba y, sobre todo, porque proponía un resultado esperado.

Gracias y buena jornada.

**Gerardo Tordoya**

Comisión 2-K

Analista Programador (Modalidad a Distancia)